

Write a program to print out the following figure.

```
*  
  
**  
  
***  
  
****  
  
*****  
  
****  
  
***  
  
**  
  
*
```

```
Module Module1  
    Sub Main()  
        Dim i, j As Integer  
  
        For i = 1 To 5  
            For j = 1 To i  
                Console.Write("*")  
            Next  
            Console.WriteLine()  
        Next  
  
        For i = 1 To 4  
            For j = 4 To i Step -1  
                Console.Write("*")  
            Next  
            Console.WriteLine()  
        Next  
  
        Console.ReadLine()  
    End Sub  
End Module
```

Write a program to print out the following figure.

1

12

123

1234

12345

```
Module Module1
    Sub Main()
        Dim i, j As Integer
        j = 1
        For i = 1 To 5
            For j = 1 To i
                Console.Write(j)
            Next
            Console.WriteLine()
        Next
        Console.ReadLine()
    End Sub
End Module
```

Write a program to print out if the entered number is prime or not.

```
Module Module1

    Sub Main()
        Dim n As Integer
        n = InputBox("Enter The Number")
        For i = 2 To 1000000000
            If n Mod i = 0 Then
                If i = n Then
                    MsgBox("Prime")
                    Exit For
                Else
                    MsgBox("Not Prime")
                End If
            End If
        Next i
    End Sub

End Module
```

```

Exit For
End If
End If
Next
End Sub

```

End Module

Write a function to calculate $c = (m! + n!) / (i! - j!)$.

Module Module1

```

Sub Main()
    Dim c, m, n, i, j As Integer
    m = InputBox("Enter The Number: ")
    n = InputBox("Enter The Number: ")
    i = InputBox("Enter The Number: ")
    j = InputBox("Enter The Number: ")

    c = (Factorial(m) + Factorial(n)) / (Factorial(i) -
Factorial(j))
    MsgBox("Result is: " & c)
End Sub

Function Factorial(ByVal x As Integer)
    Dim f As Integer
    f = 1
    For i = 1 To x
        f = f * i
    Next
    Return (f)
End Function

```

End Module

Write a program to read a tow dimensional array (a) and then find:

- *The number of even elements.*
- *The sum of the positive odd element.*

Module Module1

```

Sub Main()
    Dim i, j, m, n, sum, r As Integer

    m = InputBox("Enter number of rows in the matrix")
    n = InputBox("Enter number of columns in the matrix")

    Dim a(m, n) As Integer
    For i = 0 To m - 1
        For j = 0 To n - 1
            a(i, j) = InputBox("A(" & i + 1 & "," & j + 1 & ")")
        Next
    Next

    r = 0
    sum = 0

    For i = 0 To m - 1
        For j = 0 To n - 1
            If a(i, j) Mod 2 = 0 Then
                r = r + 1
            End If
        Next
    Next

    For i = 0 To m - 1
        For j = 0 To n - 1
            If a(i, j) Mod 2 > 0 Then
                sum = sum + a(i, j)
            End If
        Next
    Next

    MsgBox("The Number of Even Numbers is: " & r & " & The Sum of  
the Positive odd numbers is: " & sum)

    End Sub
End Module

```

Write a program to read a tow dimensional array (a) and then find:

- *The sum of each raw.*
- *The sum of each column.*
- *The product of the main diagonal.*

Module Module1

```
Sub Main()  
    Dim i, j, M, N, Row_Sum, Column_Sum, Product As Integer  
  
    M = InputBox("Enter number of rows in the matrix")  
    N = InputBox("Enter number of columns in the matrix")  
  
    Dim a(m, n) As Integer  
    For i = 0 To m - 1  
        For j = 0 To n - 1  
            a(i, j) = InputBox("A(" & i + 1 & "," & j + 1 & ")")  
        Next  
    Next  
  
    For i = 0 To M - 1  
        Row_Sum = 0  
        For j = 0 To N - 1  
            Row_Sum = Row_Sum + a(i, j)  
        Next  
        MsgBox("The Sum of the elements in the row " & i + 1 & "  
is: " & Row_Sum)  
    Next  
  
    For i = 0 To M - 1  
        Column_Sum = 0  
        For j = 0 To N - 1  
            Column_Sum = Column_Sum + a(j, i)  
        Next  
        MsgBox("The Sum of the elements in the column " & i + 1 & "  
is: " & Column_Sum)  
    Next  
  
    Product = 1  
  
    For i = 0 To m - 1  
        Product = Product * a(i, i)  
    Next  
  
    MsgBox("The Product of the elements in the main diagonal is: "  
& Product)  
End Sub  
  
End Module
```

Using the optional keyword calculate the maximum of two or three positive numbers.

Module Module1

```
Sub Main()  
    Dim Max, R As Integer  
    Dim N(3) As Integer
```

```
aaa:    R = InputBox("Press 3 or 2, according how many numbers you  
want to get the maximum of: ")
```

```
    If R > 3 Or R < 2 Then  
        MsgBox("Error")  
        GoTo aaa  
    End If
```

```
    For i = 0 To R - 1  
        N(i) = InputBox("Enter Number " & i + 1)  
    Next
```

```
    If R = 2 Then  
        Max = Maximum(N(0), N(1))  
        MsgBox(Max)  
    Else  
        Max = Maximum(N(0), N(1), N(2))  
        MsgBox(Max)  
    End If
```

```
End Sub
```

```
Function Maximum(ByVal x As Integer, ByVal y As Integer, Optional  
z As Integer = 0)  
    Dim Max As Integer
```

```
    If x > y Then  
        If x > z Then  
            Max = x  
        Else  
            Max = z  
        End If
```

```
    Else
```

```

        If y > z Then
            Max = y
        Else
            Max = z
        End If
    End If

    Return (Max)
End Function
End Module

```

Another Solution by Anwar Wagdy

Module Module1

```

Sub Main()
    Dim R, Max, N1, N2, N3 As Integer
aaa:   R = InputBox("If you want to get the maximum number of two
numbers press 2, but if you want to get the maximum number of three
numbers press 3")

    If R > 3 Or R < 2 Then
        MsgBox("Error")
        GoTo aaa
    End If

    If R = 2 Then
        N1 = InputBox("Enter the first number")
        N2 = InputBox("Enter the second number")
        Max = Maximum(N1, N2)
        MsgBox("The maximum number is " & Max)
    Else
        N1 = InputBox("Enter the first number")
        N2 = InputBox("Enter the second number")
        N3 = InputBox("Enter the third number")
        Max = Maximum(N1, N2, N3)
        MsgBox("The maximum number is " & Max)
    End If
End Sub

Function Maximum(ByVal x As Integer, ByVal y As Integer, Optional
z As Integer = 1)
    Dim Max As Integer

```

```
If x > y And x > z Then
    Max = x
ElseIf y > x And y > z Then
    Max = y
Else
    Max = z
End If

Return (Max)
End Function

End Module
```
